

1 **CLAIMS**

2

3 1. A speech system, comprising:

4 a speech engine configured to recognize commands from a user and make

5 announcements to the user;

6 a speech server having a speech server interface through which multiple

7 speech-enabled applications communicate with the speech system, and a speech

8 application programming interface through which the speech server communicates

9 with the speech engine; and

10 wherein the speech server manages concurrent processing of interactions

11 submitted by the speech-enabled applications while allowing each speech-enabled

12 application to utilize a different speech recognition grammar.

13

14 2. The speech system as recited in claim 1, wherein the speech server is

15 configured to support persistent grammars for the speech-enabled applications so

16 that a user command belonging to a persistent grammar is recognized whether the

17 associated speech recognition application is running or not.

18

19 3. The speech system as recited in claim 2, wherein the speech server is

20 further configured to launch a speech-enabled application if the speech-enabled

21 application uses a persistent grammar and the speech-enabled application is idle

22 when a recognition is made that belongs to the persistent grammar used by the

23 speech-enabled application.

24

25

1 4. The speech system as recited in claim 1, wherein the speech server is
2 further configured to:
3 support yielding grammars for the speech-enabled applications; and
4 deactivate all but one active yielding grammar. .
5

6 5. The speech system as recited in claim 1, wherein the speech server is
7 further configured to:
8 support yielding grammars and global grammars for the speech-enabled
9 applications; and
10 activate the global grammars and continuously monitor for recognitions in
11 the global grammars.
12

13 6. The speech system as recited in claim 1, further comprising a
14 grammar table for each grammar used by the speech-enabled applications, each
15 grammar table containing one or more grammar attributes for the grammar with
16 which it associated.
17

18 7. The speech system as recited in claim 6, wherein a grammar
19 attribute in each grammar table is a grammar identifier that uniquely identifies the
20 grammar associated with the grammar table.
21
22
23
24
25

1
2 12. The speech system as recited in claim 6, wherein a grammar
3 attribute in each grammar table is a static flag that, if set, indicates that the
4 grammar associated with the grammar is a static grammar that cannot be changed
5 after the grammar table is stored in the speech system.
6

7 13. The speech system as recited in claim 6, further comprising a master
8 grammar table that contains each grammar table used by the speech recognition
9 applications.
10

11 14. A method, comprising:
12 receiving a speech interaction from a speech-enabled application;
13 identifying a speech grammar associated with the speech interaction; and
14 processing the speech interaction according to grammar attributes contained
15 in a grammar table associated with the identified speech grammar.
16

17 15. The method as recited in claim 14, wherein the speech interaction is
18 a first speech interaction, the speech grammar is a first speech grammar, the
19 grammar table is a first grammar table, the speech-enabled application is a first
20 speech recognition application, and the method further comprises:

21 receiving a second speech interaction from a second speech-enabled
22 application while the first speech interaction is being processed;

23 identifying a second speech grammar associated with the second speech
24 interaction; and
25

1 processing the second speech interaction according to grammar attributes
2 contained in a second grammar table that is associated with the second speech
3 grammar.
4

5 16. The method as recited in claim 15, wherein the processing the
6 second speech interaction further comprises processing the second speech
7 interaction after the first speech interaction has concluded processing if a grammar
8 attribute in the second grammar table indicates that the second speech grammar is
9 a yielding grammar.
10

11 17. The method as recited in claim 14, wherein a grammar attribute
12 included in the grammar table is a grammar identifier that uniquely identifies the
13 speech grammar associated with the grammar table.
14

15 18. The method as recited in claim 14, wherein a grammar attribute
16 included in the grammar table is an executable command that, when executed,
17 launches the speech recognition application associated with the grammar table.
18

19 19. The method as recited in claim 14, wherein a grammar attribute
20 included in the grammar table is a global flag that, when set, indicates that the
21 speech grammar associated with the grammar table is a global grammar that may
22 only be deactivated by the speech-enabled application that submitted the speech
23 interaction and, when not set, indicates that the speech grammar associated with
24 the grammar table is a yielding grammar.
25

1 20. The method as recited in claim 14, wherein a grammar attribute
2 included in the grammar table is a persistent flag that, when set, indicates that the
3 speech grammar associated with the grammar table is a persistent grammar that is
4 active even when the speech recognition application associated with the speech
5 grammar is not loaded.

6
7 21. The method as recited in claim 14, wherein a grammar attribute
8 included in the grammar table is an active flag that, when set, indicates that the
9 speech grammar associated with the grammar table is currently active and, when
10 not set, indicates that the speech grammar associated with the grammar table is
11 currently inactive.

12
13 22. The method as recited in claim 14, wherein a grammar attribute
14 included in the grammar table is a static flag that, when set, indicates that the
15 speech grammar associated with the grammar table may not be changed after the
16 speech grammar is committed.

17
18 23. One or more computer-readable media containing computer-
19 executable instructions that, when executed on a computer, perform the following
20 steps:

21 receiving a first interaction from a first speech-enabled application that
22 utilizes a first grammar;

23 processing the first interaction according to properties attributable to the
24 first grammar;

1 receiving a second interaction from a second speech-enabled application
2 that utilizes a second grammar, the second interaction being received while the
3 first interaction is processing;
4 processing the second interaction according to properties attributable to the
5 second grammar;
6 wherein the processing of the first interruption is interrupted and processing
7 of the second interruption is immediately commenced if an indication is detected
8 directing that the second interaction be processed immediately.

9
10 24. The one or more computer-readable media as recited in claim 23,
11 wherein the indication directing that the second interaction be processed
12 immediately is a flag contained in the interaction.

13
14 25. The one or more computer-readable media as recited in claim 23,
15 wherein the indication directing that the second interaction be processed
16 immediately is a property attributable to the second grammar.

17
18 26. The one or more computer-readable media as recited in claim 23,
19 further comprising providing a grace period after processing of the second
20 interaction has completed before beginning processing a third interaction.

21
22 27. The one or more computer-readable media as recited in claim 23,
23 further comprising completing processing of the first interaction after the
24 processing of the second interaction has concluded.
25

1 28. The one or more computer-readable media as recited in claim 23,
2 further comprising:
3 detecting a self-destruct indicator associated with the first interaction; and
4 terminating further processing of the first interaction.
5

6 29. A speech server interface exposed by a speech system for use by one
7 or more speech-enabled applications, comprising a persist method that a speech-
8 enabled application uses to persist a grammar used by the speech-enabled
9 application.
10

11 30. The speech server interface as recited in claim 29, wherein the
12 persist method further comprises a launch path parameter that provides an
13 executable command that the speech system uses to launch the speech-enabled
14 application.
15

16 31. A speech server interface exposed by a speech system for use by one
17 or more speech-enabled applications, comprising a create grammar method that is
18 used to load an existing grammar used by a speech-enabled application or to load a
19 new grammar for the speech-enabled application.
20

21 32. The speech server interface as recited in claim 31, further
22 comprising a remove grammar method that is used to remove a grammar from the
23 speech system.
24
25

33. A speech server interface exposed by a speech system for use by one or more speech-enabled applications, comprising a yield-to-grammar method used by a speech-enabled application to de-activate all yielding grammars in the speech system other than the grammar used by the speech-enabled application.

34. The speech server interface as recited in claim 33, further comprising an unyield-to-grammar method used by a speech-enabled application to make other yielding grammars unyield.

35. A speech server interface exposed by a speech system for use by one or more speech-enabled applications, comprising an advise speech events method used by a speech-enabled application to let the speech system know that the speech-enabled application is listening for speech recognition events.

36. The speech server interface as recited in claim 35, further comprising an unadvise speech events method used by a speech-enabled application to let the speech system know that the speech-enabled application is not listening for speech recognition events.

37. The speech server interface as recited in claim 36, further comprising:

a create grammar method that is used to load an existing grammar used by a speech-enabled application or to load a new grammar for the speech-enabled application;

1 a get grammar identifier method that is used to obtain a value uniquely
2 associated with the loaded grammar;

3 a remove grammar method that is used to remove a grammar from the
4 speech system;

5 a persist method that a speech-enabled application uses to persist a
6 grammar used by the speech-enabled application;

7 a yield-to-grammar method used by a speech-enabled application to make
8 yielding grammars in the speech system yield to the grammar used by the speech-
9 enabled application;

10 an unyield-to-grammar method used by a speech-enabled application to
11 make other yielding grammars unyield;

12 a commit method used to commit a grammar to the speech system;

13 a get rule method used by the speech system to construct and control
14 individual rules in a grammar;

15 a create new state method used by the speech system to create new states in
16 a grammar;

17 an add word transition method used to add a transition between two states
18 on a word;

19 an add rule transition method used to add a transition between two states on
20 a rule;

21 a set rule state method used to activate and de-activate rules;

22 a set grammar state method used to activate and de-activate grammars;

23 a get grammar state method used to get a grammar state;

24 a get recognition method used to get a recognition that has occurred;

1 a get alternate method used to get alternates to a recognition that has
2 occurred;
3 a turn speech recognizer on method that is used by a speech-enabled
4 application to activate a speech recognizer in the speech system;
5 a turn speech recognizer off method that is used by a speech-enabled
6 application to deactivate a speech recognizer in the speech system;
7 a get recognizer state method used to get a speech recognizer state;
8 an advise SAPI event method used to pass in a sink that is called when an
9 event that is advised for occurs;
10 an unadvise SAPI event method used to let the speech system know that a
11 sink used with the advise SAPI event method is no longer interested in SAPI
12 events;
13 a get recognition context method used to get a speech recognition context
14 pointer from a speech engine in the speech system; and
15 a get voice method used to get a voice pointer from the speech engine.
16
17
18
19
20
21
22
23
24
25